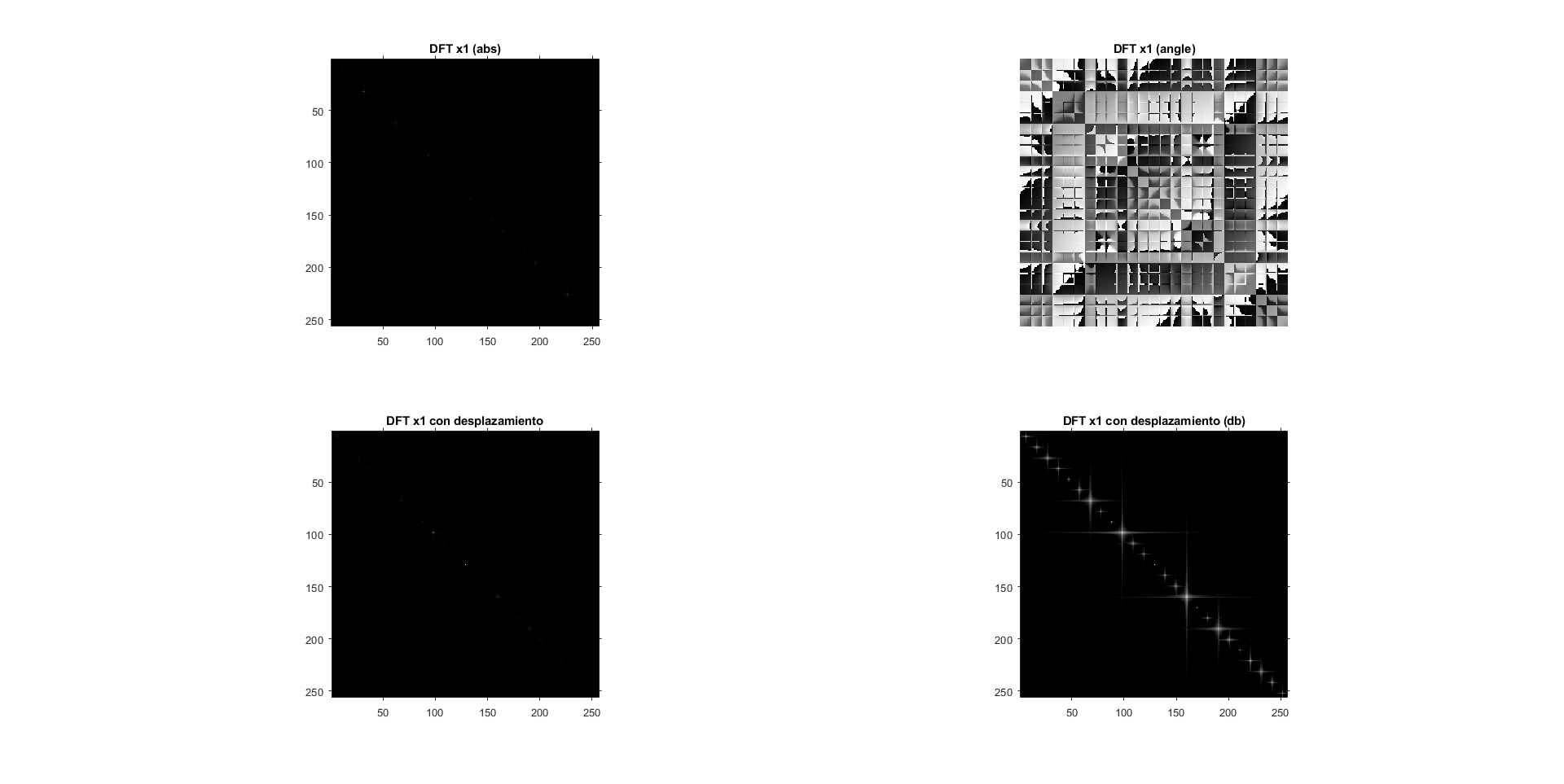
|  |  |
| --- | --- |
|  | **UPV, Facultad de Informática**  **Dpto. de Arquitectura y Tecnología de Computadores** |
| **PROCESADO DIGITAL DE SEÑAL** | |
| PROYECTO ESPECÍFICO - LABORATORIO 5 (FIm)  **ANALISIS FRECUENCIAL: IMAGEN** | |
| Componentes del grupo: - Alex Beltran  - Daniel Cañadillas  - Ainhoa Serrano | |
| **Nota**: Enviar este documento “Lab5\_FIm\_resultados.doc” completado con las tareas solicitadas, el código generado y los comentarios y aclaraciones que consideréis oportunos, junto con los correspondientes ficheros .m, en un fichero .zip vía eGela. | |

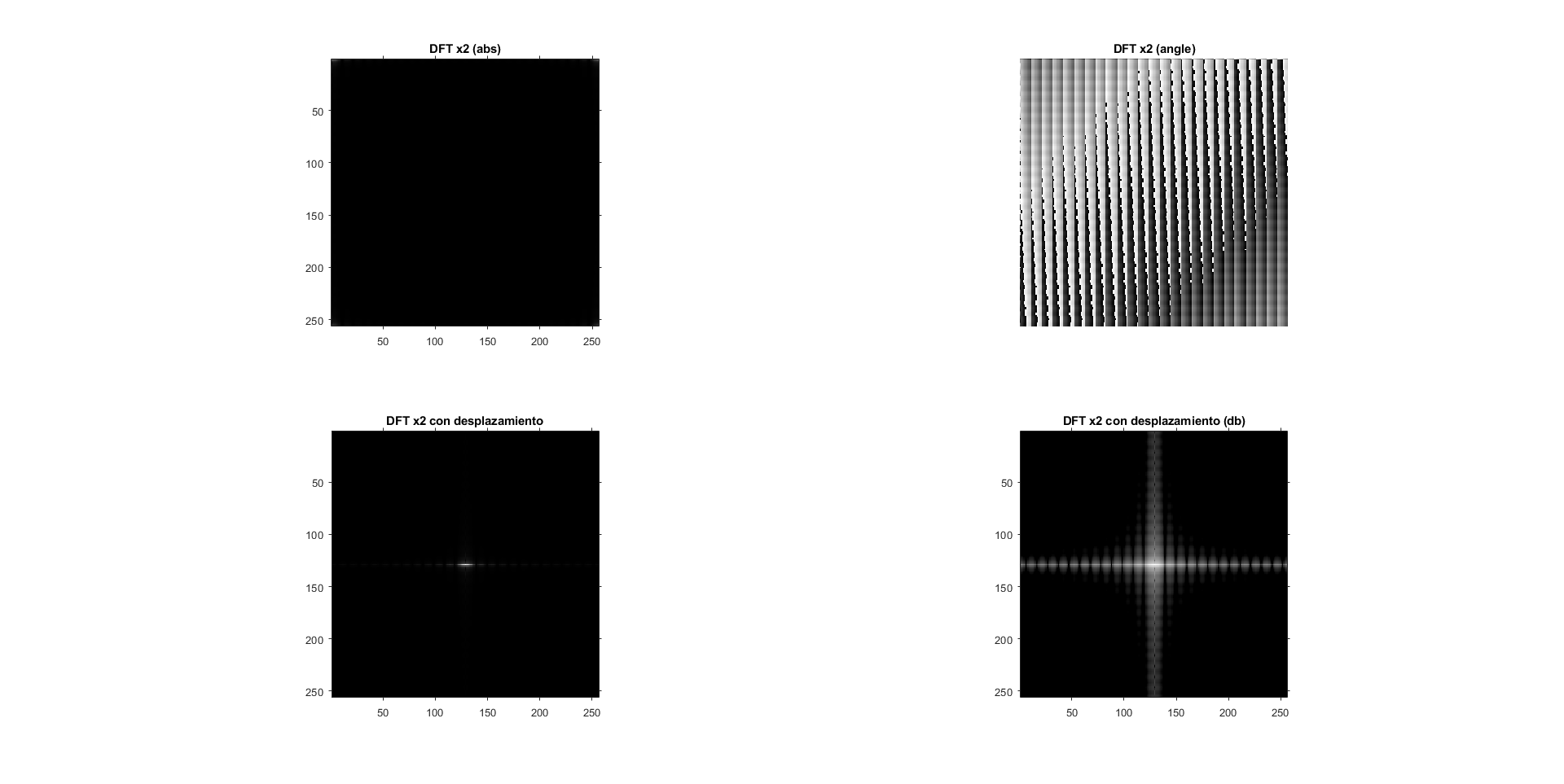
**DFT PARA SEÑALES BIDIMENSIONALES (P5\_1\_dft\_bidim\_ej1\_ej2.m)**

**Ejercicio 1**:

1. Genera las figuras de los espectros de magnitud (desplazado y en db) y de los espectros de fase de las dos señales x1, y x2 que se proponen. Incluye el código utilizado para generar las figuras.

|  |
| --- |
| [m, n] = ndgrid(0:255, 0:255);    x1 = (1/24) \* rem(3\*m+3\*n,25);  x2 = (m < 25 & n < 25);    % DFT sin desplazamiento  X1 = fft2(x1);  X2 = fft2(x2);    % DFT con desplazamiento  X1s = fftshift(X1);  X2s = fftshift(X2);    % DFT en db  X1db = 20\*log10(abs(X1s));  X2db = 20\*log10(abs(X2s));    % plot x1  figure;  subplot(2,2,1);  imshow(abs(X1),[]),axis on;  title('DFT x1 (abs)');  subplot(2,2,2);  imshow(angle(X1),[]);  title('DFT x1 (angle)');  subplot(2,2,3);  imshow(abs(X1s),[]),axis on;  title('DFT x1 con desplazamiento');  subplot(2,2,4);  x1dbmax = max(X1db(:));  imshow(X1db,[x1dbmax-60 x1dbmax]),axis on;  title('DFT x1 con desplazamiento (db)');    % plot x2  figure;  subplot(2,2,1);  imshow(abs(X2),[]),axis on;  title('DFT x2 (abs)');  subplot(2,2,2);  imshow(angle(X2),[]);  title('DFT x2 (angle)');  subplot(2,2,3);  imshow(abs(X2s),[]),axis on;  title('DFT x2 con desplazamiento');  subplot(2,2,4);  x2dbmax = max(X2db(:));  imshow(X2db,[x2dbmax-60 x2dbmax]),axis on;  title('DFT x2 con desplazamiento (db)'); |

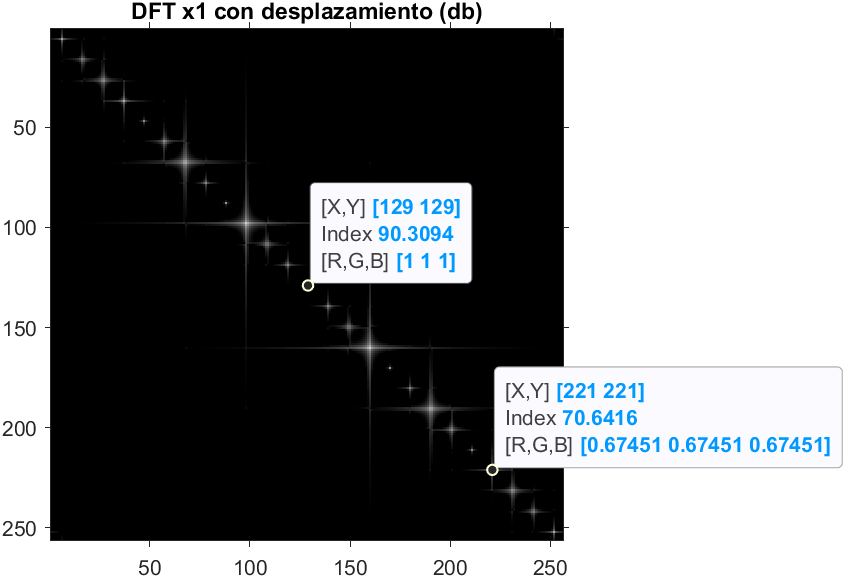
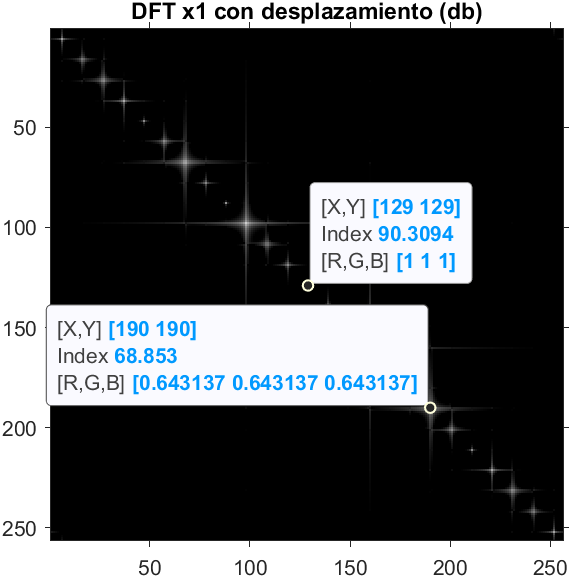
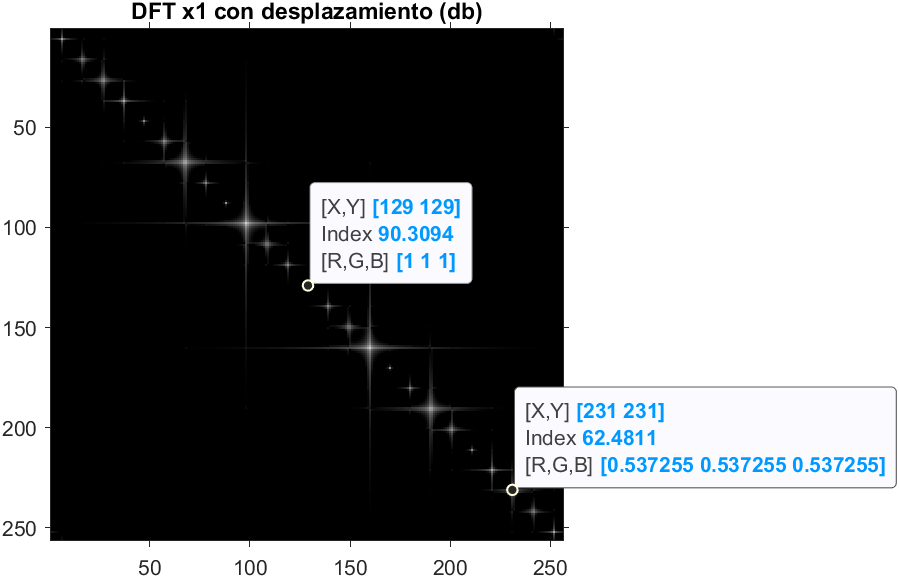




1. En el caso de x1 calcula las frecuencias asociadas a la frecuencia fundamental y al menos a tres armónicos distintos con mayor contribución. Explica el método seguido y adjunta las figuras empleadas.

|  |
| --- |
| % Frecuencia fundamental X1  % Al estar shifteado el centro se encuentra en (129,129)  % El punto que se corresponde a la frecuencia fundamental se encuentra en  % el (160,160)  % Por lo que (160-129,160-129) = (31,31) -> (32,32)  % La frecuencia fundamental para ambos componentes se calcularia como  F\_x1 = 31/256; % Aproximadamente 3/25 Hz    % Armónicos  %(190,190)  %(190-129,190-129)  F\_x1\_1 = 61/256;    % (231,231)  % (231-129,231-129)  F\_x2\_2 = 102/256;    % (221,221)  % (221-129,221-129)  F\_x2\_3 = 92/256; |

Como se puede apreciar se han obtenido las coordenadas relativas al centro de la imagen shifteada (e.g. (190-129,190-129) ), una vez obtenida dicha imagen se ha pasado el proceso inverso del cálculo de sus coordenadas, es decir, a las coordenadas relativas al centro se divide la dimensión de dicha coordenada y se obtiene la frecuencia. Como ambas coordenadas son iguales las frecuencias son las mismas en ambas direcciones.



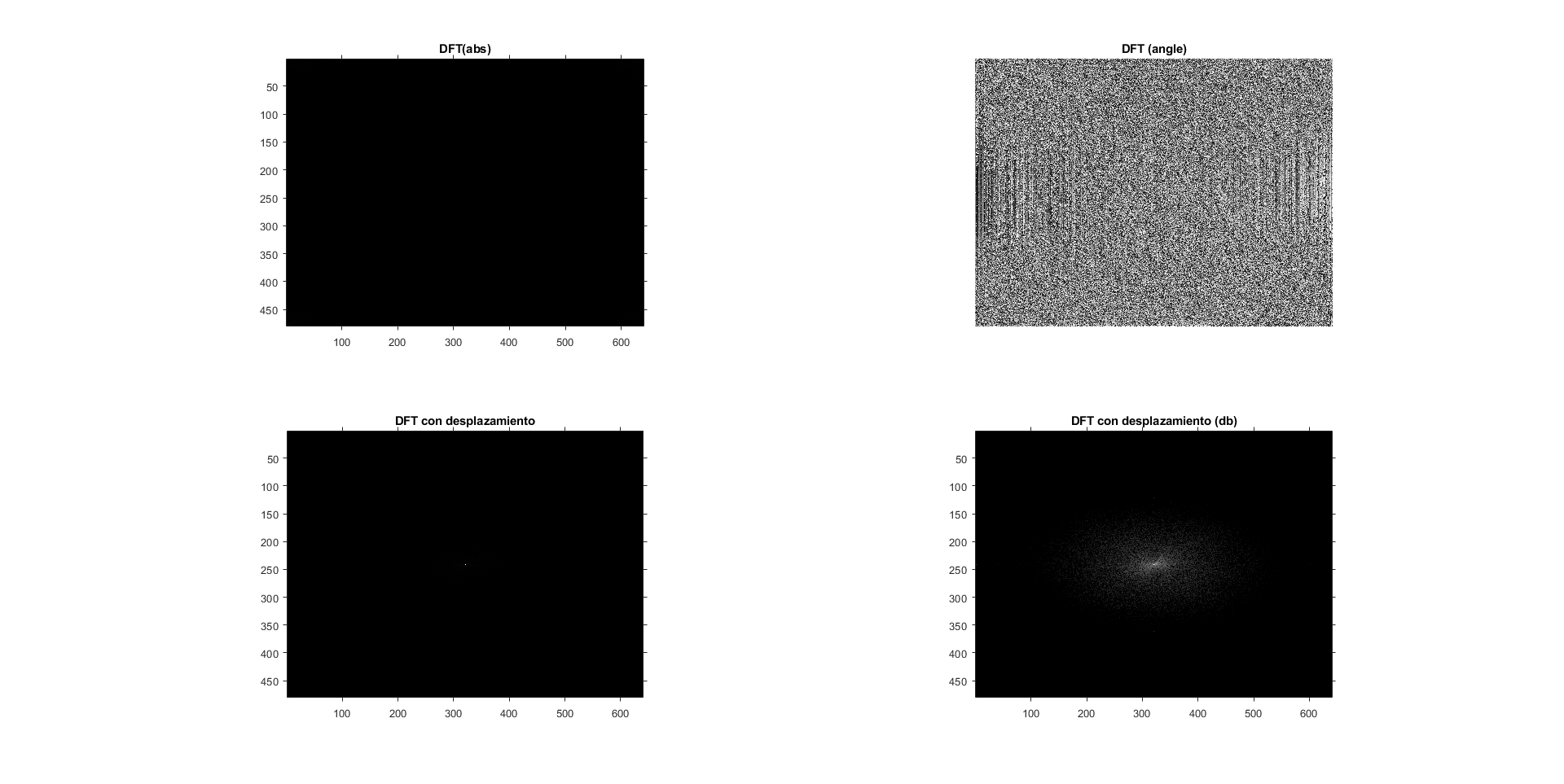
**Ejercicio 2**:

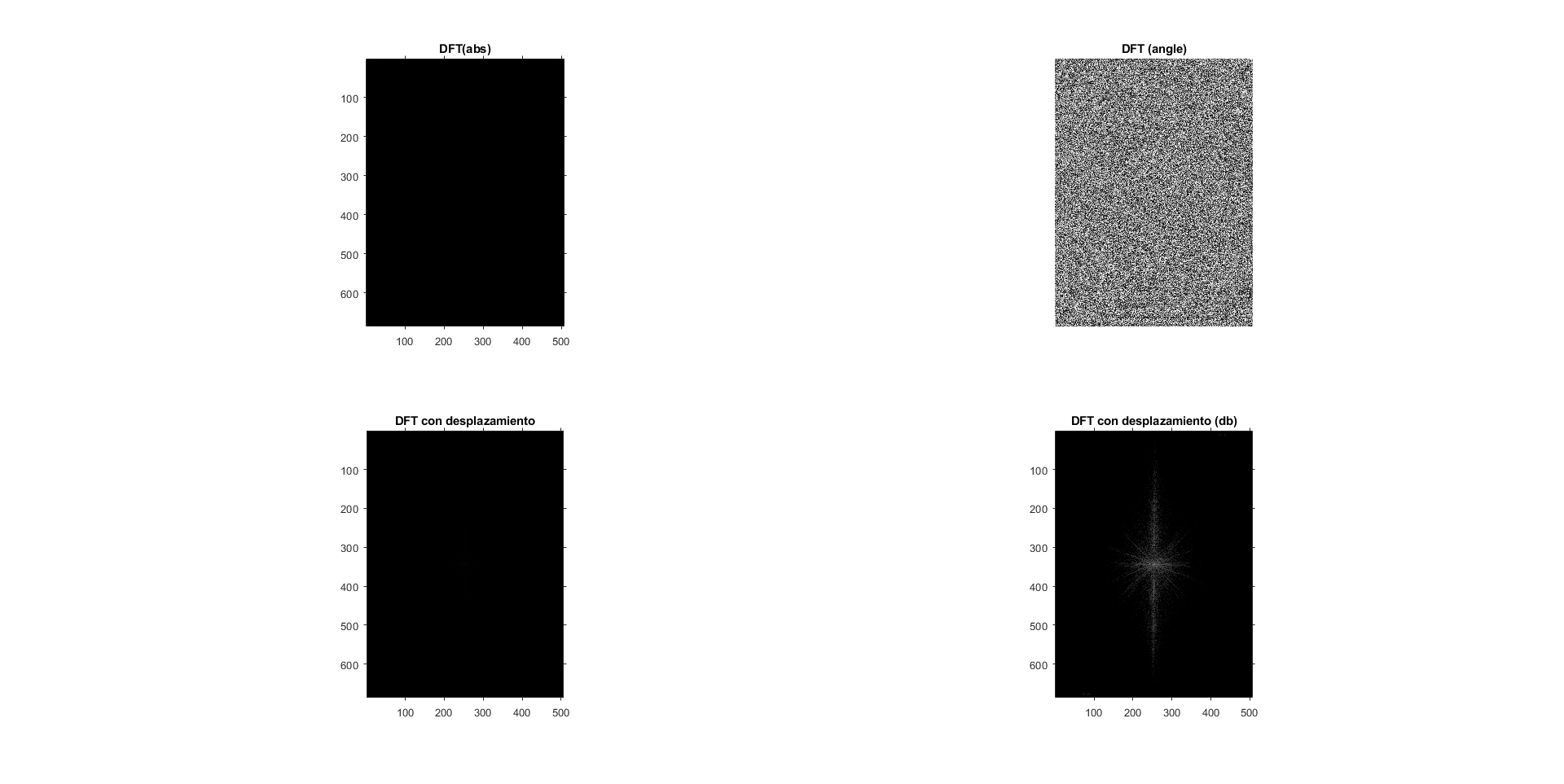
1. Genera las figuras de los espectros de magnitud (desplazado y en db) y de los espectros de fase de las dos imágenes. Incluye el código utilizado para generar las figuras. ¿Qué se observa en los espectros de magnitud de estas dos imágenes?

|  |
| --- |
| % ----------------- Ejercicio 2 -----------------------  X = imread('arboles.jpg');  XX = fft2(X);  XXs = fftshift(XX);  XXdb = 20\*log10(abs(XXs));      figure;  subplot(2,2,1);  imshow(abs(XX),[]),axis on;  title('DFT(abs)');  subplot(2,2,2);  imshow(angle(XX),[]);  title('DFT (angle)');  subplot(2,2,3);  imshow(abs(XXs),[]),axis on;  title('DFT con desplazamiento');  subplot(2,2,4);  xdbmax = max(XXdb(:));  imshow(XXdb,[xdbmax-60 xdbmax]),axis on;  title('DFT con desplazamiento (db)');  X = imread(‘vias.jpg');  XX = fft2(X);  XXs = fftshift(XX);  XXdb = 20\*log10(abs(XXs));      figure;  subplot(2,2,1);  imshow(abs(XX),[]),axis on;  title('DFT(abs)');  subplot(2,2,2);  imshow(angle(XX),[]);  title('DFT (angle)');  subplot(2,2,3);  imshow(abs(XXs),[]),axis on;  title('DFT con desplazamiento');  subplot(2,2,4);  xdbmax = max(XXdb(:));  imshow(XXdb,[xdbmax-60 xdbmax]),axis on;  title('DFT con desplazamiento (db)'); |

Como se puede observar, no se percibe ninguna frecuencia dominante dentro del espectro de magnitud debido al origen de las imágenes (no generadas mediante sinusoides o señales periódicas).

* Árboles

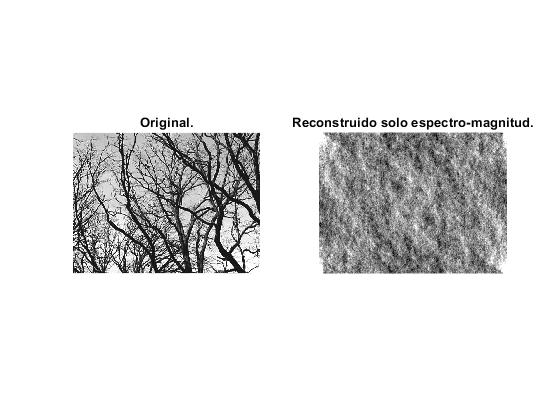


* Vías

**IDFT**

**Ejercicio 3**: **(P5\_2\_ifft2\_magnitud\_ej3.m)**

Adjunta la imagen restituida sin tener en cuenta el espectro de fase. ¿Qué ocurre? Incluye el código utilizado para generar las figuras.



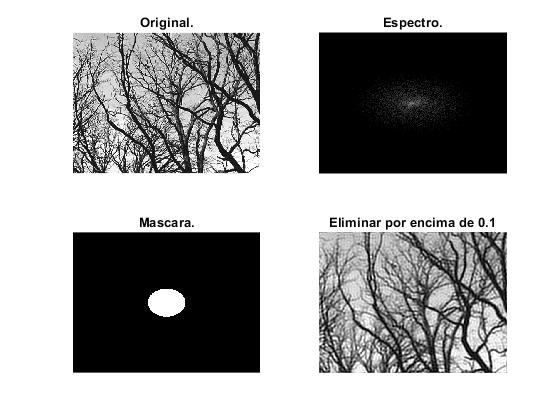
**Al no tener en cuenta el espectro de fase, perdemos gran parte de la información, acabando con una imagen borrosa.**

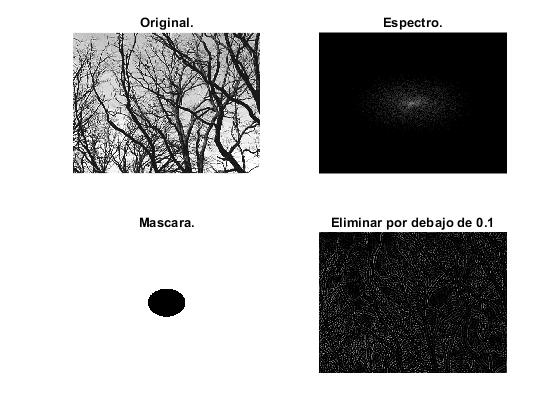
|  |
| --- |
| %% Ejercicio 3    %% Codigo:  [arbol] = imread('arboles.jpg');  fftArbol = fft2(arbol);  shiftedArbol = fftshift(fftArbol);  espectro\_magnitud = ifft2(ifftshift(abs(shiftedArbol)));    %% Visualizacion:  figure,    subplot (1,2,1), imshow(arbol);  title('Original.');    subplot (1,2,2), imshow(uint8(real(espectro\_magnitud)));  title('Reconstruido solo espectro-magnitud.'); |

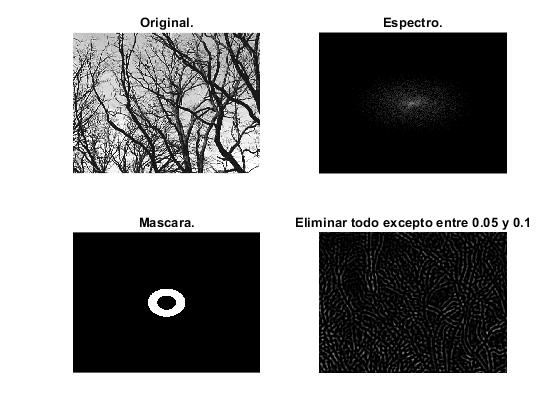
**Ejercicio 4**: **(P5\_3\_mascaras\_ej4.m)**

Utilizando el procedimiento mascara elimina de ‘**arboles.jpg**’ las frecuencias a) por encima de 0.1, b) por debajo de 0.1 y c) las que están fuera de una franja entre 0.05 y 0.1. Visualiza los resultados en cada caso: la imagen original, el espectro de la imagen original, la máscara generada y la imagen final obtenida, incluyendo el código utilizado para generarlas. Comenta los resultados.

**Al enmascarar las imágenes, realmente lo que estamos haciendo es pasar las imagenes for filtros (o convoluciones) pasa-alta y pasa-baja, que eliminan ciertas frecuencias de las imágenes, resaltando “features” como los bordes o contornos ( omo se puede ver en las últimas dos imágenes) o simplificando la imagen borrando las hojas ( como pasa en la primer caso ).**







|  |
| --- |
| clc; clear all; close all;  [arbol] = imread('arboles.jpg');  fftArbol = fft2(arbol);  X = fftshift(fftArbol);  Xdb = 20\*log10(abs(X));  dbmax = max(Xdb(:));    %% Eliminar Por encima de 0.1  M = mascara(size(X), 0.1, 'bajo');  Xf = M.\*X;  xf = uint8(real(ifft2(ifftshift(Xf))));  figure,  subplot (2,2,1), imshow(arbol), title("Original.");  subplot (2,2,2), imshow(Xdb, [dbmax-60, dbmax]), title("Espectro.");  subplot (2,2,3), imshow(M), title("Mascara.");  subplot (2,2,4), imshow(xf), title("Eliminar por encima de 0.1");    %% Eliminar Por debajo de 0.1  M = mascara(size(X), 0.1, 'alto');  Xf = M.\*X;  xf = uint8(real(ifft2(ifftshift(Xf))));  figure,  subplot (2,2,1), imshow(arbol), title("Original.");  subplot (2,2,2), imshow(Xdb, [dbmax-60, dbmax]), title("Espectro.");  subplot (2,2,3), imshow(M), title("Mascara.");  subplot (2,2,4), imshow(xf), title("Eliminar por debajo de 0.1");    %% Eliminar fuera de 0.05 y 0.1  M1 = mascara(size(X), 0.05, 'alto'); % Quitamos por debajo de 0.05  Xf = M1.\*X;  M2 = mascara(size(X), 0.1, 'bajo'); % Quitamos por encima de 0.1  Xf = M2.\*Xf;  xf = uint8(real(ifft2(ifftshift(Xf))));  figure,  subplot (2,2,1), imshow(arbol), title("Original.");  subplot (2,2,2), imshow(Xdb, [dbmax-60, dbmax]), title("Espectro.");  subplot (2,2,3), imshow(M1.\*M2), title("Mascara.");  subplot (2,2,4), imshow(xf), title("Eliminar todo excepto entre 0.05 y 0.1"); |